UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/998,848	11/15/2001	Kenneth Y. Ogami	CD01177M	6884

60909 7590 06/21/2010
CYPRESS SEMICONDUCTOR CORPORATION
198 CHAMPION COURT
SAN JOSE, CA 95134-1709

EXAMINER	
VO, TED T	

ART UNIT	PAPER NUMBER
2191	

MAIL DATE	DELIVERY MODE
06/21/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 09/998,848	**Applicant(s)** OGAMI, KENNETH Y.
	Examiner TED T. VO	**Art Unit** 2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *10 February 2010*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-14,16,26-30 and 36-38* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) *26-30* is/are allowed.

6)☒ Claim(s) *1-14,16,36 and 37* is/are rejected.

7)☒ Claim(s) *38* is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed on 02/10/2010, entered by the request filed on 03/02/2010.

Claims 1 and 13 are amended. Claims 1-14, 16, 26-30, 36-38 are pending in this application.

Response to Arguments

2. Applicants' arguments in the Remarks section filed on 08/12/2009 have been respectfully considered. The amendment of claims 1 and 13 fails to put the claims into the allowable condition because the features which are added into claims 1 and 13 are only in conformity with a virtual block in an IDE.

All the claims will be allowed if the independent claims include all the features of the allowed claim 26.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

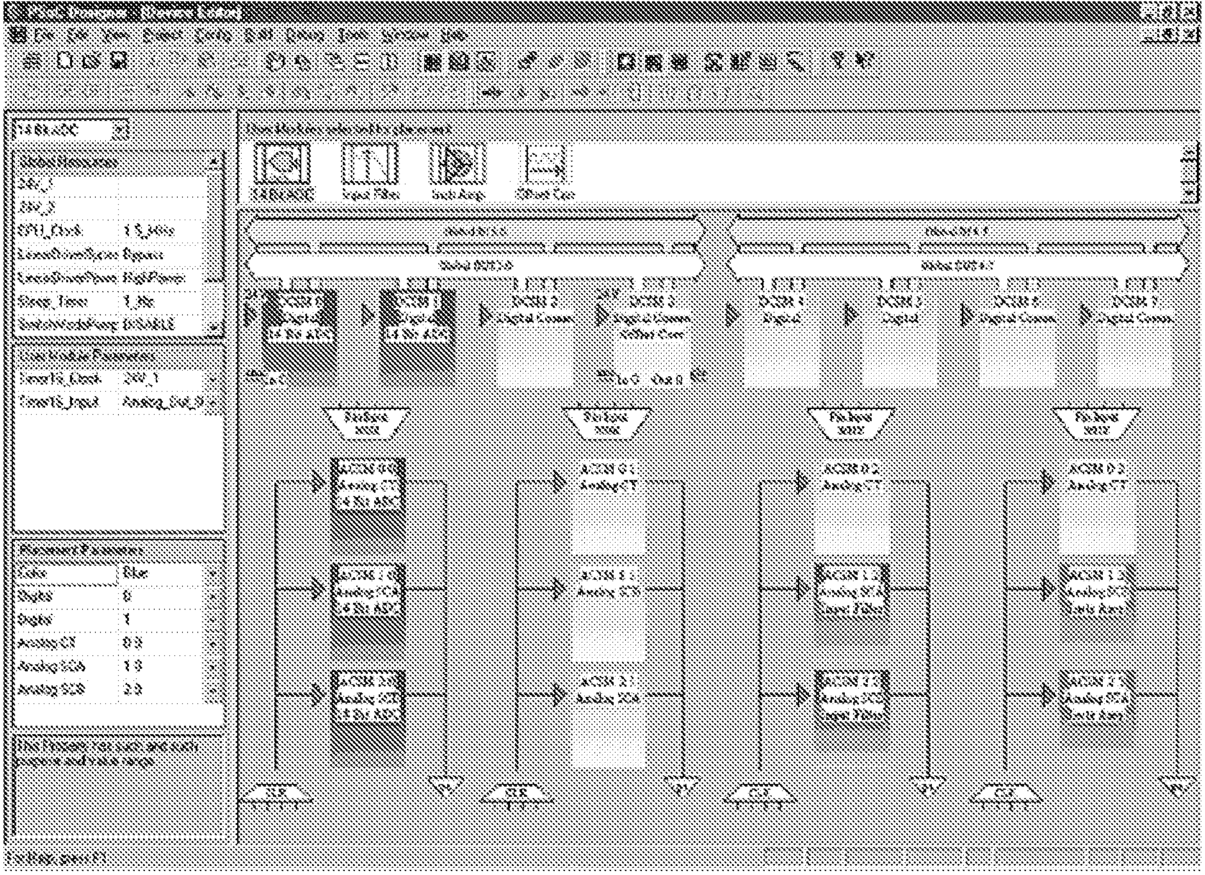
A person shall be entitled to a patent unless –

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-14, 16, 36-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bindra, “Programmable SoC Delivers A New Level Of System Flexibility”, 2000, in view of Hamblen, “Rapid Prototyping using Field Programmable logic Devices”, 6-2000.

As per Claim 1: Bindra discloses

Figure 4



4. The PSoC Designer is an integral part of the Windows-based development process. Its device editor employs a graphical interface to connect user modules, which are next mapped onto the SoCblocs on-chip. Finally, the user selects the pin assignments.

Which includes displaying a collection of virtual blocks in a design system, each virtual block (i.e. a block in main pane) is a programmable block that is selected from the top pane for embedding to the collection for forming a project. With a user module being connected to a virtual block, configured by corresponding information provided in the left pane, it forms a user module presenting the functional circuit of a virtual block. The figure 1 appears a building block that is embedded within the collection because it contains input/output pins connecting to another virtual block; and thus, if depict a selected block such as a module of Figure 1 or another simpler block, it will provide a user to construct the code module representing the function of the depicted block. It appears covers the limitations recited within:

A method for configuring a microcontroller, comprising:

displaying a first graphical user interface on a display device of a computer system, said first graphical user interface comprising a collection of virtual blocks in a design system (see figure 4);

receiving at said computer system a selection of a user module, wherein said user module comprises information for implementing a function using a programmable physical block (Figure 4, selecting a 14 Bit ADC, part of Library modules or user modules (p. 3, include line 5));

displaying on said display device a second graphical user interface operable for receiving user-specifiable information about said user module (Figure 4, selecting a 14 Bit ADC as a virtual block, and its module (p. 3) provided with user input parameters in the left area of the PSoC Designer)***;***

assigning a virtual block taken from said collection to said user module, wherein said virtual block corresponds to said programmable physical block (Figure 4, selecting a 14 Bit ADC, part of Library modules or user modules (p.3), and take the input parameters assigned by user into the module);

With regards to,

constructing ~~completely~~ computer-generated source code, ~~that is loaded into a register of said programmable physical block to cause said programmable~~

~~*physical block to implement said function*~~*, wherein constructing the computer-generated source code comprises* ~~*substituting said user-specifiable information for generic information in a template file*~~ *automatically constructing assembly code holding configuration information for said programmable physical block, wherein said configuration information is based on said user-specifiable information and comprises information that is loaded into a register of said programmable physical block to cause said programmable physical block to perform said function.*

Bindra shows Figure 4 which displays SoCblocks. Each of the blocks is interconnected with a user module; where a user module is about the programming elements to implement the function of a SoCblock (Figure 4). The PSoC includes toolbars with specific icons and commands. For Example, "Config', Debug, etc. Thus, it is clearly that the PSoC is an IDE which is able to execute the module or either to compile the module into an executable code.

See (p.1),

> "[A]ccording to Dennis Seguin, principal applications engineer at the company, this is aided by a PC-based integrated design environment that's tightly coupled to the PSoC architecture. "Using this high-level development software, a user can quickly configure and reconfigure analog and digital arrays of the device on a PC screen. And upon satisfaction, the user can then map the system configured onto the PSoC chip by the click of an icon," Seguin remarks.
> By allowing the user to work with menus and graphical icons on-screen, this Windows-based IDE permits a user to drag and drop higher-level functions in appropriate boxes and create the end system in minutes. Once the system configuration is finalized, it's stored in the flash memory on-chip. Upon power-up, the contents of the flash are transferred into register space that holds the configuration information. Also, this system configuration can be easily and quickly modified as the system requires change.

Bindra does not specific mention the configuration information (appeared being programming elements) as that, "*assembly code holding configuration information for said programmable physical block*".

However, it should be noted that information and modules that implement virtual blocks are source code under the execution of the IDE must be as assembly language or executable code. There is no thing new for this.

Hamblen shows a programmable on chip design process (p. 36, Figure 11) using a CAD tool that takes design virtual blocks to be generated in form of assemble or machine language. It is automatically generated by a C compiler to mapped on to the Virtual blocks designed from the CAD tool (See Figure 1: VHDL Design entity (i.e. Virtual blocks) connected with automatically generated code generated by a C Compiler, mapped to virtual blocks at gate levels in the CAD tool) for obviously disclosing: "*automatically constructing source code*"

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teaching of Hamblen. With the suggestion of Bindra about the IDE, configuration, debug, etc, the module in the PSoC Designer (discussed by Bindra), the teaching about the assembly code of Hamblen must be included. Thus, in conformity with a virtual block of an IDE, the programming element in the module must be a type of source code or assembly *code holding configuration information for said programmable physical block,* i.e. it would be conforming for a standard tool of an IDE (i.e. PSoC Designer, shown by Bindra). It is also obvious because, in order to present the functionality of a circuit, it will be advantage for the user to use a tool. Otherwise, it will **take days** for the user to manually assemble a program of a circuit/block assigned with parameters presenting element configuration.

As per Claim 2: Bindra further discloses, *The method of Claim 1, wherein said function comprises a pulse width modulator* (Bindra: See Figure 4, "User Module" that represents various Digital functions, and see P.2 line 36, "PWMs").

As per Claim 3: Bindra further discloses, *The method of Claim 1, wherein said function comprises a timer.* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 36, "*timers*").

As per Claim 4: Bindra further discloses, *The method of Claim 1, wherein said function comprises an analog-to-digital converter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 35, "ADCs").

As per Claim 5: Bindra further discloses, *The method of Claim 1, wherein said function comprises a digital-to-analog converter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 35 "DACs").

As per Claim 6: Bindra further discloses, *The method of Claim 1, wherein said function comprises a counter* (Bindra: See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 36 "counters").

As per Claim 7: Bindra further discloses, *The method of Claim 1, wherein said function comprises a signal amplifier.* (See Figure 4, refer to "User Module" that represents various Digital functions, and see P.2 line 33 *"differential amplifiers*").

As per Claim 8: Bindra further discloses, *The method of Claim 1, wherein said function provides serial communication.* (See Figure 4, refer to "User Module" that represents various Digital functions, and see P.3, line 9, "serial transmitters/receivers").

As per Claim 9: Bindra further discloses, *The method of Claim 1, wherein said collection is displayed as a two dimensional array of programmable analog virtual blocks and programmable digital virtual blocks.* (See collections in the right bottom section, which is *two-dimensional array*).

As per Claim 10: Bindra further discloses, *The method of Claim 1, wherein said assigning further comprises assigning a second virtual block to said user module* (See collections in the right bottom section, which is two *dimensional array*).

As per Claim 11: Bindra further discloses, *The method of Claim 1, wherein said source code comprises a symbolic name for a register address in said programmable physical block.* (Bindra: See page 2, lines 12-17 ('register space that holds the configuration information').

As per Claim 12: Bindra further discloses, *The method of Claim 11 wherein said symbolic name is derived from said function.* (See Bindra 'User module" in Figure 4, where user module represents a circuit element. Each circuit element is a symbolic name function: e.g.: ADC, DAC, Timer, Counter, etc).

As per Claim 37: Bindra further discloses, *The method of Claim 1 wherein said user module is represented by first markup language data that includes information defining how configuration register for said microcontroller are be programmed in order to implement said function, and wherein said programmable physical is represented by second markup data that includes information defining physical addresses of said configuration registers.*

Official notice is taken that Markup language such as SGML, HTML, XML, is the language that is provided computer rendering. There is no invention on using the markup language for rendering, but compliant to the rules set forth by the language. This is well known, The Figure 4 of Bindra can be implemented by XML and its data, and the use of Markup languages is only to take the advantage of the availability that is designed for graphical rendering (The well-known HTML and XML is available in the Internet, Example, Applicants can refer to the website XML.com).

Therefore, it is obvious for an ordinary in the art when something like mark up language become standard for providing the graphical rendering; one will take the advantage to use it by including it for conforming to the availability.

To support this office notice: see Karayiannis "Using XML for Representation and Visualization of Elaborated VHDL-AMS Models", 2000, IEEE, pages 83-87.

As per Claim 36: Incorporated to the rejection of claim 1, Bindra and Hamblen further discloses

The method of Claim 1 wherein said constructing the computer-generated source code further comprises:

reading the template file;

producing assembly, include, and header files from the template file,

wherein said user-specifiable information comprises information specific to said user module, information specific to said function

and information specific to a control parameter of said function;

compiling said assembly, include and header files to produce an executable file; downloading said executable file as a code block to a memory of said microcontroller; and executing said code block to configure said programmable block.

as in the Figures 7, 9-11 (Hamblen). It is obvious, because the claimed recitation conforms to or complies with basis process of C compiler when it generates assembly language; i.e. a C program has “include statements”, or “header files”; therefore, every template created from the C compiler for an assembly program will include with “header files” such #include statement on its top. It is obvious because it is conforming to a C program → assembly program.

Bindra and Hamblen further in combined fail to disclose a very well-known markup language, that is usually available for graphical rendering; where the Markup language is usable for rendering virtual blocks such as the blocks shown in Figure 4.

Official notice is taken: the obviousness is as addressed as set forth in the rejection in the claim 37.

To support this office notice:
see Texas Instruments “TMS320C6000 Optimizing Compiler User’s Guide”, 4-2001, Texas Instruments, chapters 1: pages 1-7 and chapter 2: pages 1-44.
see Miguel “Implementation of an universal Boot Monitor for an
ARM-based System”, 5-2000, TU Berlin, Germany, chapter 4: pages 53-100.
As per Claim 13: See the rationale addressed in Claim 1.

As per Claim 14: Regarding,

“The method of Claim 13, wherein said automatically constructing further comprises:

computing a register address for a register within said programmable physical block;

determining a symbolic name for said register address, said symbolic name corresponding to said user module and said circuit design; and substituting said symbolic name for a generic name in said template assembly code”. See page 2, lines 12-17 (‘register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish');

- *computing a register address for a register within said programmable block:* page 6, lines 7-13, referring "register mapping"

- *determining a symbolic name for said register address, said symbolic name corresponding to said user module and said circuit design:* page 2, lines 12-17, referring "holds the configuration information".

- *substituting said symbolic name for a generic name in said template assembly code:* referring the code construction performed by the PSoC Designer.

As per Claim 16: regarding limitations of Claim 16.

See page 2, lines 12-17 ('register space that holds the configuration information') and page 6, lines 7- 13, ('user modules are selected, pins are assigned, and register mapping are establish') for

- *determining a symbolic name corresponding to said user module and said circuit design;* referring "holds the configuration information".

- *computing a register address for a register within said programmable block;* referring "register mapping"

- *assigning said symbolic name to said register address; and placing said symbolic name into said assembly code in place of a generic name provided in said template assembly code file:* referring the code construction performed by the PSoC Designer.

Allowable Subject Matter

5. Claims 26-27 are allowed.

6. Claim 38 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ted T. Vo whose telephone number is (571) 272-3706. The examiner can normally be reached on 8:00AM to 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708.

The facsimile number for the organization where this application or proceeding is assigned is the Central Facsimile number 571-273-8300.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

TTV
June 19, 2010

/Ted T. Vo/
Primary Examiner, Art Unit 2191